NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

GRAPHIC SIMULATION OF AN
AIR DEFENSE SCENARIO

by

Klaus Schuster

September 1987

Thesis Advisor: James D. Esary

Approved for public release; distribution is unlimited

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION	UNCLASSIFIED
1b RESTRICTIVE MARKINGS	
2a SECURITY CLASSIFICATION AUTHORITY	
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
3 DISTRIBUTION / AVAILABILITY OF REPORT	Approved for public release; distribution is unlimited.
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	
5 MONITORING ORGANIZATION REPORT NUMBER(S)	
6a NAME OF PERFORMING ORGANIZATION	Naval Postgraduate School
6b OFFICE SYMBOL (If applicable)	55
7a NAME OF MONITORING ORGANIZATION	Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code)	Monterey, California 93943-5000
7b ADDRESS (City, State, and ZIP Code)	Monterey, California 93943-5000
8a NAME OF FUNDING / SPONSORING ORGANIZATION	
8b OFFICE SYMBOL (If applicable)	
9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c ADDRESS (City, State, and ZIP Code)	
10 SOURCE OF FUNDING NUMBERS	PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO
11 TITLE (Include Security Classification)	Graphic Simulation of an Air Defense Scenario
12 PERSONAL AUTHOR(S)	SCHUSTER, Klaus
13a TYPE OF REPORT	Master's Thesis
13b TIME COVERED	FROM ____ TO ____
14 DATE OF REPORT (Year, Month, Day)	1987 September
15 PAGE COUNT	69
16 SUPPLEMENTARY NOTATION	

17 COSATI CODES		
FIELD	GROUP	SUB-GROUP

18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Graphic, Simulation, Basic-language, Air Defense, Combat Model

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

This paper presents the development of an animated graphical computer simulation of an air defense scenario. In particular, the air defense scenario consists of a variable number of attackers, pictured as airplanes and a variable number of defenders, pictured as air defense artillery tanks. The simulation allows choosing between two different target assignment protocols. Protocol 1 simulates a random target assignment, where every defender selects a target at random to shoot at. Protocol 2 simulates a controlled target assignment, where the defenders are assigned to a target and continue to track the target until a kill is made. The simulation is written in BASICA, uses an IBM Personal Computer (IBM PC) and requires a color monitor and a color/graphics adapter card.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT	☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS
21 ABSTRACT SECURITY CLASSIFICATION	UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL	Prof. James D. Esary
22b TELEPHONE (Include Area Code)	408-646-2780
22c OFFICE SYMBOL	55Ey

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

Approved for public release; distribution is unlimited.

Graphic Simulation of an Air Defense Scenario

by

Klaus Schuster
Captain, Federal Rep. of Germany, Air Force
Diplom Ingenieur (FH) der Elektrotechnik
Hochschule der Bundeswehr, Muenchen, 1978

Submitted in partial fulfillment of the
requirement for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1987

ABSTRACT

This paper presents the development of an animated graphical computer simulation of an air defense scenario. In particular, the air defense scenario consists of a variable number of attackers, pictured as airplanes and a variable number of defenders, pictured as air defense artillery tanks. The simulation allows choosing between two different target assignment protocols. Protocol 1 simulates a random target assignment, where every defender selects a target at random to shot at. Protocol 2 simulates a controlled target assignment, where the defenders are assigned to a target and continue to track the target untill a kill is made. The simulation is written in BASICA, uses an IBM Personal Computer (IBM PC) and requires a color monitor and a color/graphics adapter card.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

Microcomputers are used more and more frequently for the simulation of all kinds of statistical processes. In military studies different combat situations which do not follow standard stochastic processes have been simulated. Simulations are also often used to manifest numerical results, even when a nice mathematical closed form could have been derived.

This paper describes the development of a computer simulation for an air defense scenario. In a typical computer simulation mathematical and logical relationships are used to create a real world model. In this development an animated graphical simulation was chosen to demonstrate the possibility of not only running a simulation and giving several numbers as an output, but also showing the actual occurrence of the battle as animated pictures on the screen. This gives the user direct insight into what is happening in the simulation without many hard-to-understand explanations.

The intent was not to solve a very tricky combat simulation, but rather to show some of the possibilities of what a computer simulation can look like.

This paper is one of several efforts at the Naval Postgraduate School to present various stochastic processes as animated graphical simulations, using an IBM Personal Computer (IBM PC-XT). Prior developments were: the simulation of several Poisson Process Models [Ref. 1], the simulation of a Machine-Repairman Model [Ref. 2] and the simulation of a Jackson Queue Network Model [Ref. 3]. These developments were a close guide and presented many ideas for this paper to follow in a series of animated graphical simulations for educational purposes.

II. THE AIR DEFENSE SCENARIO MODEL

A. DISCUSSION

The simulation of the model of an Air Defense Scenario, described in this paper, contains a simple Force-on-Force scenario, where several defenders, pictured as Air Defense Artillery (ADA), intercept several attackers, pictured as Airplanes.

The defenders are stationary on the ground and are actually drawn on the screen as ADA Tanks [Figure 1]. The user can choose between one and seven defenders.

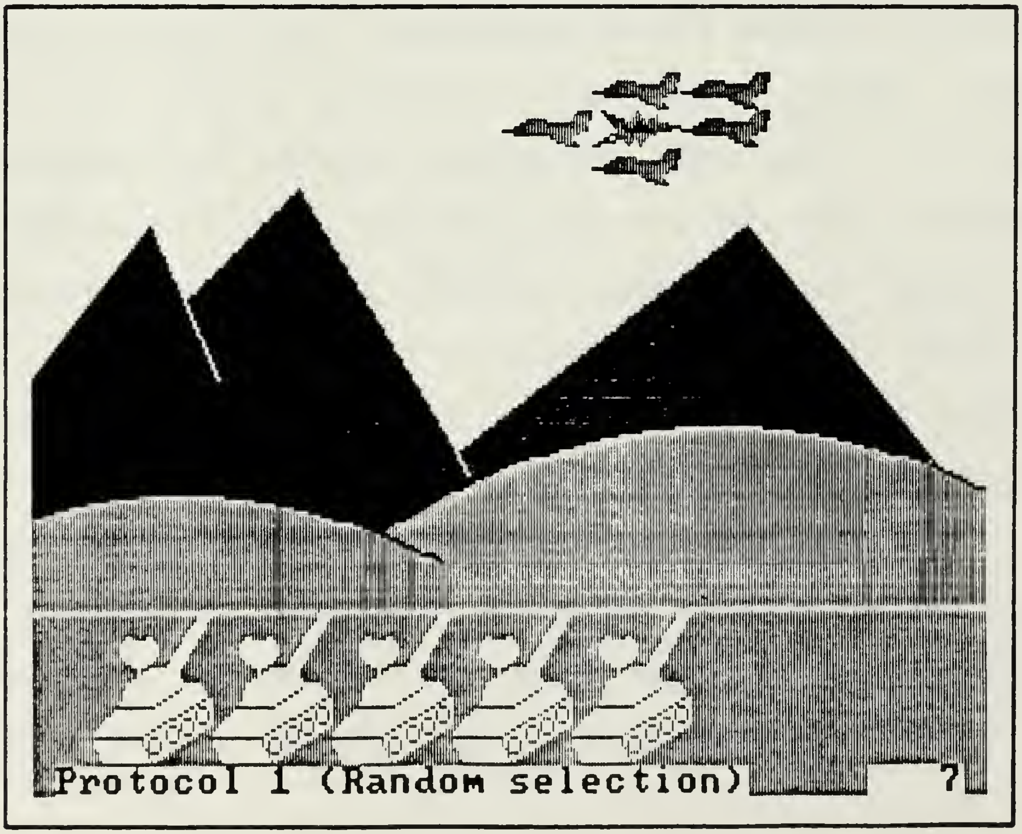

Figure 1. Main Screen

The attackers fly from right to left in the top-section of the screen and are drawn as an airplane formation, for which the user can choose between one and nine Airplanes.

The simulation model actually consists of two Air Defense scenarios with two different target assignment protocols.

1. <u>Assignment Protocol 1</u>

Protocol 1 is the simulation of a random target assignment, where the defender randomly chooses a target for every shot he fires. It is meant to be a demonstration of the real world situation, where no communication among the defenders exists and all defending resources are used to shoot. It will be discovered that there are a number of multiple hits on the same aircraft because of the lack of a controlled target assignment. This will reduce the percentage of kills per shot, especially when there are more defenders shooting and there are less attackers to intercept as well as when the hit probability for every defender is high.

2. <u>Assignment Protocol 2</u>

Protocol 2 is the simulation of a controlled target assignment, where each defender is assigned to a different target and holds on to it until a kill is made. After hitting his target, the defender is given a

new target assignment. This is done until there are no unassigned targets remaining. Having only as many defenders shooting as there are targets remaining will result in saving defense resources. There will be no multiple assignments and, therefore, no multiple hits. The percentage of kills per shot will, in the long run, reach the hit probability chosen for the simulation. It reflects the real world situation where there is communication between the defender and the command control for target assignments.

3. Model Assumptions

Both protocols are simulated and run independently, as chosen by the user, and are shown on the same type of screen [Figure 1]. Several assumptions are made to simplify the model:

- After a target formation has shown up on the screen, the time elapsed from detecting the targets to the actual shot is equal for all defenders. The user can enter this as the **acquisition time** at the beginning of the simulation run. The time during which the attackers are exposed to the defenders is also constant and can be determined by the user as the **exposure** time. This means that, when the acquisition time is higher than or equal to the exposure time, no shot can be fired by the defenders and all airplanes will survive.

- Having fired at a target there is no delay before the defender is able to fire again, even when there is a new target assignment. This is not quite true in the real world, but the incorporation of a delay would have slowed down the graphic simulation and the result would have been the same. This is because the simulation runs only in relative time units.
- When an airplane is hit for the first time, it is considered a **kill**. A second hit within the same time unit is not counted as a kill, but as a **multiple hit**.
- The purpose of a simulation is to estimate some arameters or the relationships between parameters. Therefore, the simulation is repeated several times and data is collected to estimate the real world relationships. This simulation model executes one attack after the other without any delay between the attacks in order to allow the simulation to be run the desired number of attacks without unnecessary intervention by the user.

B. MODEL ESTIMATORS

The following paragraph explains the model estimators, how they are defined and how they are calculated from the simulation parameters.

After the completion of the desired number of attacks the program will show a list of numerical or graphical statistics. They are based on:

- the number of surviving airplanes
- the number of kills made
- the number of multiple hits made
- the number of shots fired by the defenders
- the number of attacks actually flown
- the number of time units from the first appearance of the attackers until the last airplane in the particular formation was killed.

1. Average Number of Shots per Attack

The number of shots which are fired at the attacker between the end of the acquisition time and the end of the attack (when all attackers are killed or the exposure time has expired) are counted. To be able to compare this number to other simulation iterations, it is not printed as the total number of shots, but instead as the average number of shots per attack and is calculated as:

$$\text{Avg \# of shots/attack} = \frac{\text{total \# of shots}}{\text{\# of attacks}}$$

The higher this number is, the less effective the defense is. It will take on values between the # of attackers and the # of defenders * (exposure time-acquisition time), i.e. 1 to 1400.

2. Average Number of Kills per Attack

The average number of kills in every attack is also a direct measurement of the effectiveness of the defense. It is calculated as:

$$\text{Avg \# of kills/attack} = \frac{\text{\# of hits} - \text{\# of mult. hits}}{\text{\# of attacks}}$$

It will be 0, when no attacker is killed and will be equal to the # of attackers when the defense has its highest effectiveness.

3. Average Number of Multiple Hits per Attack

In cases where an airplane is hit more than once by the defenders within the same time unit, the airplane is killed by the first hit. All additional hits are considered as multiple hits. This is only possible under protocol 1 (random assignments), because in protocol 2 (controlled assignments) multiple hits are specifically avoided. It is computed as:

$$\text{Avg \# of multi. hits/attack} = \frac{\text{\# of multiple hits}}{\text{\# of attacks}}$$

This average number depends greatly on the number of attackers and defenders as well as on the desired hit probability. Clearly, as more defenders shoot against less attackers with a higher hit probability, the higher the number of multiple hits will be.

4. Average Number of Survivors per Attack

Establishing the number of surviving airplanes at the end of their exposure time will give a direct measurement of the effectiveness of the defense. In order to compare this number with other attacks, the average number of survivors per attack is computed as:

$$\text{Avg \# of survivors/attack} = \frac{\text{\# of survivors}}{\text{\# of attacks}}$$

The values are between zero and the number of attackers.

5. Average Number of Kills per Shot

Related to the theoretical probability of hitting an attacker is the computed average number of kills per shot. In the case of protocol 2, these numbers should be similar. But in the case of protocol 1, where multiple hits will occur, the average number of

kills per shot will be less than the probability of hit. It is computed as:

$$\text{Avg \# of kills/shot} = \frac{\text{\# of hits - \# of add. hits}}{\text{\# of shots}}$$

The values will range from zero to approximately the theoretical probability of hit.

6. <u>Average Time until the Last Attacker is Killed</u>

The time unit in which the last attacker was killed is the actual end of the attack. It is recorded as the acquisition time plus the time the attack lasts. The minimum value would be the acquisition time and the maximum value would be the exposure time.

7. <u>Percentage of Successful Interceptions</u>

The opposite of the average number of survivors per attacker is the average number of successful interceptions. It is derived as follows:

$$\text{Avg \# of interceptions} = 1 - \frac{\text{\# of survivors}}{\text{\# of attackers}}$$

Multiplying the Average number of interceptions by 100 gives the percentage of successful interceptions. This value ranges from 0 to 100.

III. THE SIMULATION

A. THE COMPUTER

The computer used for this graphic simulation is an IBM Personal Computer (IBM-PC) or a 100% IBM-PC compatible clone. In order to run the simulation, one of these computers must be equipped with a Color/Graphics adapter, a color monitor, at least 1 floppy disk drive, and at least 128k-bytes of random access memory (RAM). All computers must be able to use IBM Advanced Basic (BASICA) in MS-DOS or PC-DOS, which requires the IBM BASIC ROM chip set. If the computer is not equipped with these chips, it must be capable of running a generic Version of BASICA, such as GW-BASIC from Microsoft. The simulation runs at a microprocessor clock speed of 4.77 MHz, 6 MHz, 8 MHz or even higher. On computers with a higher clock speed, the simulation will run faster, which will shorten the required simulation run time when large numbers of attacks are desired. The simulation program, AIRDEF, was written using the MS-DOS disk operating system Version 3.2, but any MS-DOS or PC-DOS Version developed after Version 2.1 will work. However, the simulation directly interacts with the disk operating system and, therefore, no guarantee may be given, that the simulation

will run on other than IBM-PC or 100% IBM-PC compatible computers.

B. THE PROGRAMMING LANGUAGE

AIRDEF is written in IBM Advanced Basic (BASICA), which is the programming language supplied with PC-DOS or MS-DOS. Other Versions like GW-BASIC from Microsoft will also work. BASICA, the language version used is a strictly interpreted language and is, therefore, much slower than compiled languages, such as TURBO BASIC from Borland or QuickBASIC from Microsoft. To follow the previous developments of graphical simulation packages at the Naval Postgraduate School the interpreted language BASICA was preferred. This is due to the fact that speed is not important in this simulation. The user needs to be able to follow the events on screen easily.

The RND Function in BASICA does not satisfactorily produce random numbers [Ref. 2]. Therefore, the subroutine RNGEN.SRT, written in 8088 assembly language, was used to produce the random numbers needed for this simulation. During the development of the simulation program, it was discovered that the uniformity of the random numbers, generated with RNGEN.SRT, improves when generating more than approximately 30 random numbers, compared to the built-in RND Function using the default

seed. Due to the fact that normally more than 50 random numbers will be generated, even in a short run of the simulation, the RNGEN.SRT was used. A subroutine to change the random number generator seed of RNGEN.SRT is also implemented in the simulation program and is easily accessible from the main menu.

C. THE PROGRAM STRUCTURE

The program listing for AIRDEF.BAS is supplied in Appendix B. The program is structured in several subroutines, to allow an easy pursuing of the steps taken in the simulation. The subroutines and modules are:

- BASICA and Color/Graphic card check
- print Title Screen
- load RNGEN.SRT
- change random number generator seed
- print Main Menu
- default simulation parameter
- change simulation parameter
- capture keyboard response
- data for attacker screen coordinates
- print numerical statistic
- print graphical statistic
- protocol 1 sequence

- protocol 2 sequence
- exit the program

D. THE PROGRAM DESCRIPTION

The simulation of the air defense scenario is animated on the graphical display [Figure 1]. After the desired number of attacks are completed the statistics are displayed on a numerical statistics display or a graphical statistics display. It depends on the user's response as to which display is prefered for the presentation on the screen.

After the program is started, it reserves 16k bytes of the memory for variables and it dimensions the graphical arrays. A check is made whether BASICA or a compatible Basic is used and whether a color/graphics monitor adaptor card is installed correctly in the computer by checking specific flags in the memory map of the computer.

The first observed screen is the title screen and after pressing any key, the RNGEN.SRT is loaded into the memory and the main program menu will be displayed on the screen. The user then has several options:

- start with default parameters and Protocol 1
- start with default parameters and Protocol 2
- change the simulation parameters

- set the random number generator seed or
- end the program.

Some of the more administrative programming tasks have been adopted in part, or in whole, from the programs written by R.E. Nelson [Ref. 1] or G.F. Green [Ref. 2] in order to fit into the series of animated graphic simulations already available.

After the simulation has been started, the main graphics screen is drawn and an airplane (as attacker), an artillery tank (as defender), a shooting cloud and a fireball are drawn and saved into an array. Depending on the desired number of defenders and attackers, the line of defenders is drawn at the bottom of the screen, and the appropriate formation of attackers is drawn at the top of the screen and stored into an array.

Following some initial settings of attribute values of the defender/attacker assignment, the formation of airplanes starts moving from right to left. The set time units for acquisition will pass before the defenders begin shooting at their targets, in accordance with their assignment protocol 1 or 2. The speed of the formation is computed from the total exposure time and transformed to increments in pixels on the screen. The higher the speed of the formation, the more pixels the formation is advanced. The speed limitations, and, therefore, the limitations on the exposure time are

given by a resolution of 320 pixels horizontally in the medium resolution graphic mode and the width of the formation of 120 pixels. This yields a maximum of 200 pixels which the formation can travel, and, therefore, 200 time units of exposure. The width of a single attacker is 30 pixels and yields the minimum time units of the exposure time.

If protocol 1 is selected, all defenders shoot at the same time. A uniformly distributed random number between 0 and 1 is drawn for each shot and for every defender and compared to the set probability of hitting a target. The single shot of each defender is considered a hit when the random number is less than the probability of a hit. A new random number is drawn and converted to a uniform distribution between 1 and the number of attackers, for the selection of a target from the remaining attackers. If the selected target is still there, the hit is considered a kill and the target is removed from the formation by placing a fireball in its place. If the target has already been hit by a previous defender, within the same time unit, the hit is considered a multiple hit and a fireball is placed on the former coordinates of the target, to visualize the multiple hit. After storing the new formation, the time is increased by the number of units, according to the speed of the formation. If it

is determined by the random number generator that there has been no hit the time is increased immediately, without going through the above computations.

If protocol 2 is selected, a fixed defender-target assignment is issued at the beginning of the attack (e.g. defender No 1 is assigned to target No 1, defender No 2 is assigned to target No 2 and so forth). Those defenders with an assignment will shoot and, depending on a random number, the shot is determined as hit or no hit. If it was a hit, the target is killed by removing it from the screen with a fireball, and the defender gets a new assignment as long as there are unassigned targets remaining. No double assignments are issued, which is more realistic and saves defending resources. The defender is immediately ready for the next shot.

In both protocols hits are also characterized by an audible beep. The attack ends, when the formation has reached the left side of the screen or when all airplanes have been killed. The aircraft reaching the left side of the screen are counted as survivors and the time units elapsed from the beginning to the end of the attack are summed up over all attacks. These variables are used to compute the statistics at the end of the simulation.

E. THE STATISTICS SCREENS

At the end of the simulation, when all desired attacks have been flown, the program will ask whether a numerical or a graphical representation of the statistics is preferred.

1. <u>The Numerical Statistics screen</u>

The numerical statistics screen [Figure 2] contains the averages computed from the empirical values of the simulation variables. They are related to the simulation parameters, which are also printed in the top section of the screen.

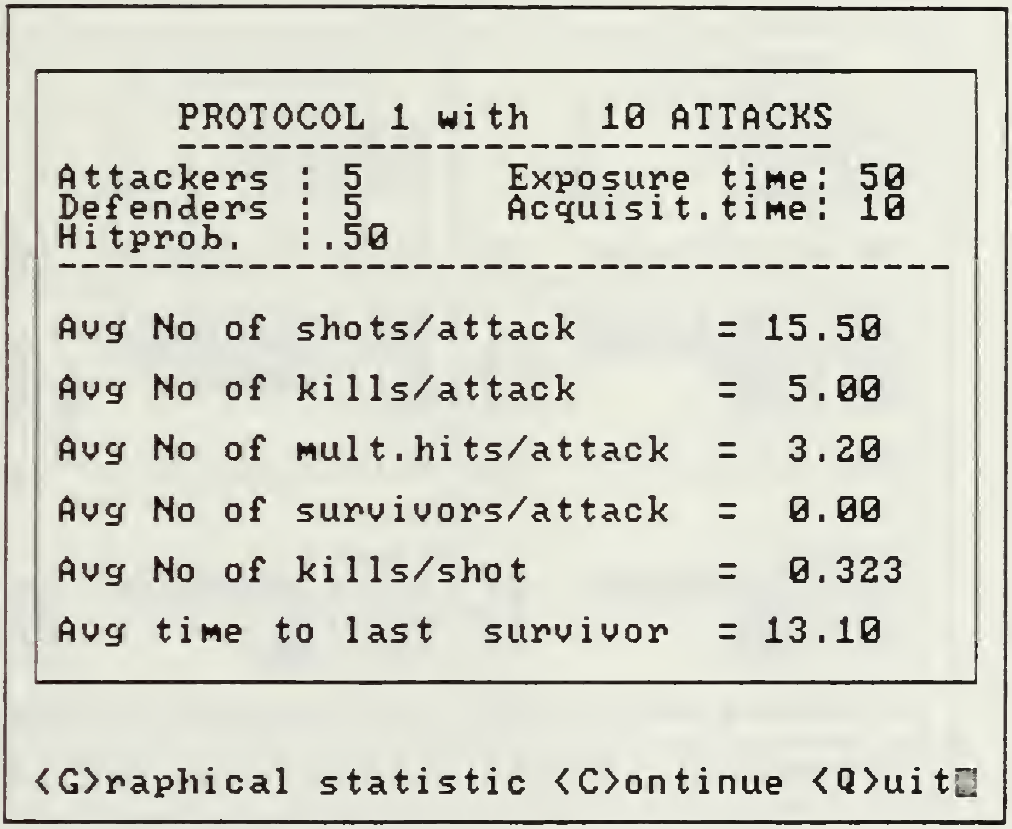

Figure 2. Numerical Statistics

The user has a choice of:

(a) switching to the graphical statistics screen

(b) continuing to the next simulation

(c) quitting the program.

2. <u>The Graphical Statistics Screen</u>

Upon switching to the graphical statistics screen [Figure 3], the numerical statistics will be presented in graphical form. It is intended to give the user a better insight into the relationships of the simulation parameters and the observed variables, by representing them in barcharts.

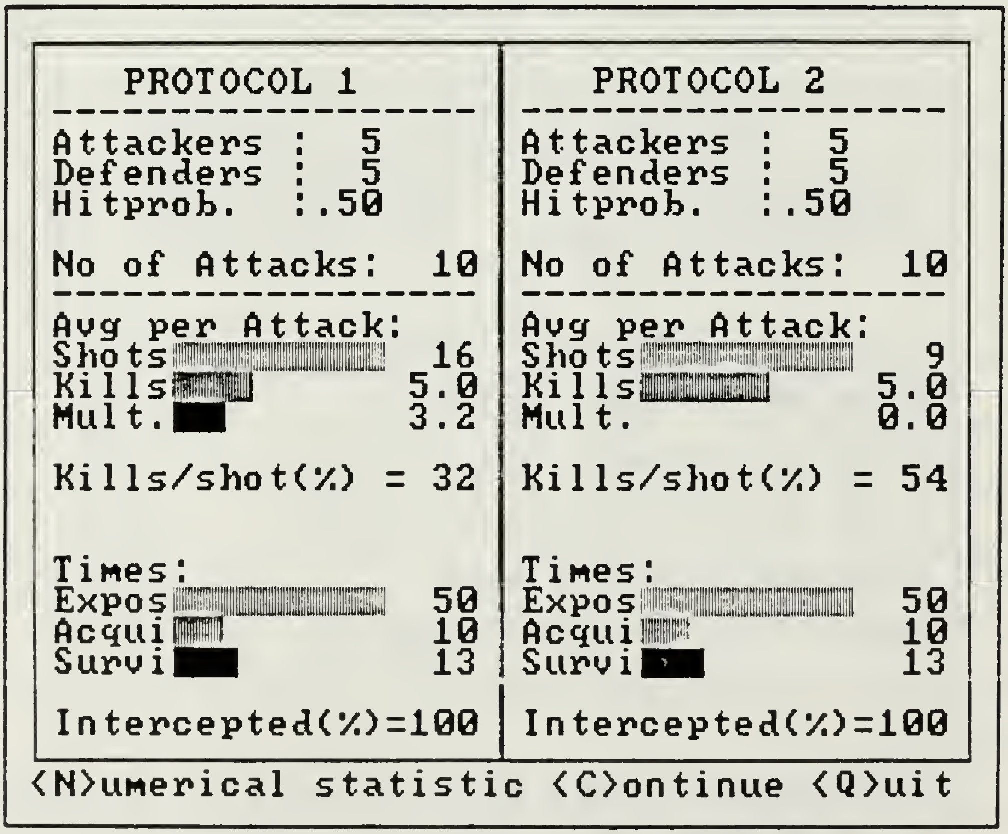

Figure 3. Graphical Statistics

When the graphical statistics screen is entered the first time, only the left half of the screen is employed. After running the simulation a second time, maybe with other parameters or using the other protocol, the former left side of the screen is moved to the right side of the screen and the new statistics are shown on the left side. This will allow an easy comparison of the new statistics with the pervious ones.

IV. CONCLUSIONS

A graphic simulation of an air defense combat scenario is presented in this paper. This animated graphical simulation allows the user to gain insight into the dynamic behavior of a combat model.

Even though the model was not intended to solve a specific type of scenario, it illustrates several conclusions as to what type of target assignment is preferable. This depends on the given simulation parameters. No communication between the defenders, as demonstrated with protocol 1, will generally result in a heavier use of defending resources, since multiple hits will occur and the defender will only stop shooting when all attackers are killed. If only the time until the last aircraft has survived is of concern and no restrictions on ammunition is made, then protocol 1 may be of a higher effectivness. But when the average number of kills per shot is important and ammunition is limited, then protocol 2 will bring better results.

To find a guideline for other specific problems, the user might want to experiment with this simulation model.

APPENDIX A

THE AIR DEFENSE SCENARIO SIMULATION

USER'S GUIDE

by

Klaus Schuster

Thesis Advisor: James D. Esary

Second Reader: Peter Purdue

TABLE OF CONTENTS

LIST OF FIGURES

A. INTRODUCTION

AIRDEF is a stochastic discrete event simulation of an air defense scenario model. It is a simple force-on-force scenario, where several defenders, depicted as Air Defense Artillery Tanks (ADA), intercept several attackers, depicted as an airplane formation. The simulation implements a variety of graphic displays to demonstrate the events in this scenario, and gives the user a choice of a numerical or a graphical display of the various statistics, computed in the simulation.

This manual is intended to assist the user in the actual utilization and operation of the program. For describing commands and keyboard entries, the following notation is used:

- ' ' single quotes are used whenever the user should type a command exactly as shown.
- < > brackets are used whenever a letter or other key should be pressed.

For Example, 'AIRDEF' <ENTER> means: type the command exactly as it appears between the quotes and press the ENTER key. PC-DOS, MS-DOS and AIRDEF make no distinction between upper or lower case letters, hence, either one may be used.

B. MODEL DESCRIPTION

AIRDEF simulates an air defense situation, where a maximum of seven Air Defense Artillery Tanks attempt to intercept a maximum of nine aircraft, flown in a formation. The purpose is to show the differences between two specific target assignment protocols, in terms of defense effectiveness.

Protocol 1 is the simulation of a random target assignment, where the defender randomly choses a target for every shot he fires. It is intended to be a demonstration of a real world situation, where no communication between the defenders or any command post exists and all defending resources are used to intercept the attackers. It will be discovered, that multiple hits on the same aircraft will occur, due to the lack of coordination in target assignment.

Protocol 2 is the simulation of a controlled target assignment, where each defender is assigned to a different target and holds on to it until a kill is made. After hitting his target the defender is reassigned a new target until there are no unassigned targets remaining. This is a simulation a real world situation, where there is communication between the defenders and a command post, which makes controlled target assignments. This will result in saving defense resources, since there are only as many defenders

shooting as there are targets remaining in the formation. Since there are no double assignments there are no multiple hits. Because of the type of target assignment the percentage of kills per shot will be higher in protocol 2 than in protocol 1.

Several assumptions are made to simplify the models:

- constant acquisition time for all defenders
- constant exposure time for all attackers
- no second acquisition time after shooting has started
- first hit is a kill
- all subsequent hits are multiple hits
- no delay between consecutive attacks

C. HARDWARE AND SOFTWARE REQUIREMENTS

1. Hardware

The simulation has been programmed for an IBM Personal Computer (IBM-PC) and uses color graphics. The computer must be equipped with a color monitor and a color/graphics adapter card, or the simulation would not function properly. The computer should have, at least, one 5 1/4 inch floppy disk drive and 128 k-bytes of random access memory (RAM). The simulation program uses BASICA, which requires 64 k-bytes, and an assembly language subroutine program, which is loaded into memory outside of BASICA.

The computer must also be able to use IBM Advanced Basic BASICA in PC-DOS or MS-DOS or to use GW-basic.

2. Software

AIRDEF is written in Advanced Basic BASICA, which is normally provided with the PC-DOS operating system. The program will also run under MS-DOS's BASICA and GW-Basic.

The primary BASIC program AIRDEF.BAS and the assembly language subroutine RNGEN.SRT are provided to the user on the distribution diskette. In addition to these programs, the user must provide the Disk Operating System PC-DOS or MS-DOS Version 2.1 (or higher) and the Advanced Basic BASICA or GW-Basic in order to run

the simulation. Since AIRDEF.BAS uses BASIC's POKE command and certain BIOS interrupt calls, there can be no guarantee, that the simulation will run properly on not 100% IBM-compatible machines.

3. Program Files

The following files are provided:

- AIRDEF.BAS....the main program in BASICA
- RNGEN.SRT.....the random number generator
- AIRDEF.BAT....to start the program in DOS
- AUTOEXEC.BAT..to auto-start the program

The first two files are necessary to run the simulation. The two batch files allow the user to either start the simulation once the DOS environment is already loaded (AIRDEF.BAT), or to auto-start the simulation while the computer is turned on. These two files are not necessary, but are for the convenience of the user. A source code listing of RNGEN.SRT is provided in Reference 2.

The user must provide the PC-DOS or MS-DOS operating system and BASICA or GW-Basic.

D. GETTING STARTED

1. Making an Application Disk

Before working with the simulation program, it is recommended that an application disk be created and the original distribution disk be stored in a safe place. Even so there are many ways to create an application diskette, this manual suggests a bootable application diskette, which may be found more convenient for single drive computers.

To make a bootable diskette follow the PC-DOS manual and format a blank diskette by typing at the DOS prompt: 'format a:/s' with the DOS diskette in drive A. The system will prompt on the screen to place a new diskette into drive A and to hit <ENTER>. The user should make sure that there is a new diskette in drive A before the <ENTER> key is pressed, otherwise the DOS diskette would be erased. The /s option causes DOS to transfer the necessary hidden files and the COMMAND.COM file to the new diskette to make it self starting by turning on the computer with the disk in drive A. The Advanced Basic file, called BASICA.COM should be transferred by the user from the DOS diskette with the COPY command. Finally, the COPY command should be used to transfer all other files from the simulation distribution diskette to the new application diskette.

2. Starting the Simulation

If the computer is turned off, place the new application diskette into drive A and turn the system on. The simulation should start automatically, when the AUTOEXEC.BAT file and all necessary program files are on the disk.

If the computer is already turned on and the A> prompt is on the screen, simply type 'AIRDEF' and the file AIRDEF.BAT will load BASICA into the memory of the computer and the simulation program AIRDEF.BAS will be loaded into the BASICA environment and automatically started.

If, for any reason, the batch files should not be utilized, the simulation can also be started by typing 'BASICA' at the DOS A> prompt and from the Basic environment by typing 'load"a:airdef' <ENTER> and 'run' <ENTER>.

During execution of the program, AIRDEF attempts to load the random number generator RNGEN.SRT from the default drive. If the program fails to find the file RNGEN.SRT on this drive, it will ask the user for the correct drive label, where the file can be found. Simply type in the proper drive label and press the ENTER key. This problem will never arise, when the bootable application diskette is used to start the computer.

3. Title Screen

Upon starting the simulation, the first screen seen by the user will be the title screen as shown in Figure 1. From this point on, the on-screen prompts will guide the user through the run of the simulation.

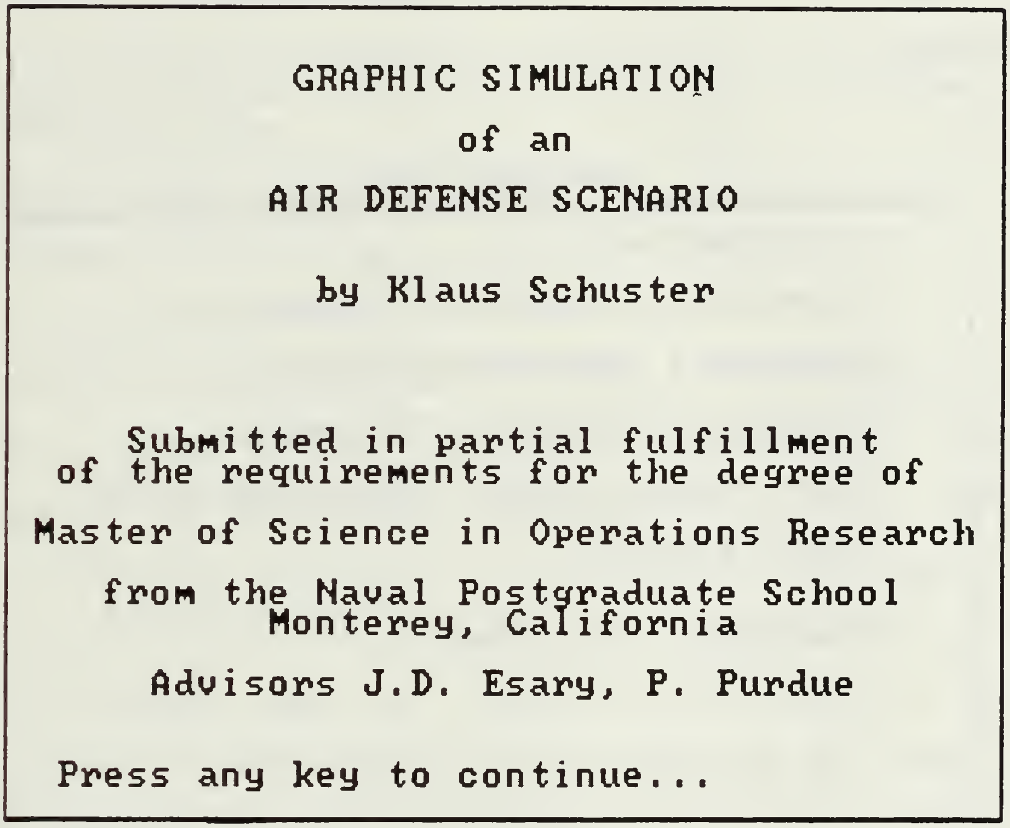

Figure 1. The Title Screen

E. PROGRAM MAIN MENU

After leaving the title screen, by pressing any key, the next screen will be the program main menu, as shown in Figure 2. The options in the program main menu allow the user to read the instructions, select the default simulation parameter for the assignment protocol 1 or 2, change the simulation parameter, get a new seed for the random number generator or end the program.

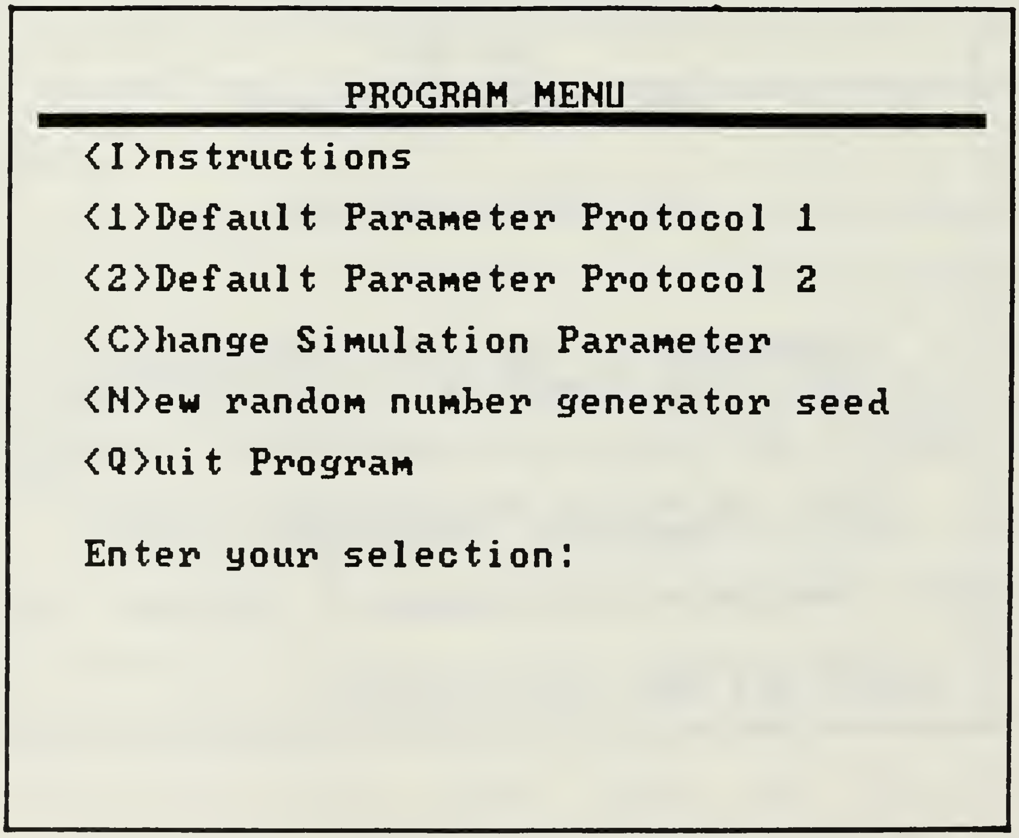

Figure 2. The Main Menu

A summarized version of the program instructions, found in this manual, are available on screen.

1. <u>Default Parameter Protocol 1</u>

If this option is selected by pressing <1>, then the simulation starts with the default values of the simulation parameters. Protocol 1 is the random target assignment protocol.

These values are:

- number of attackers = 7
- number of defenders = 5
- probability of hit = .2
- exposure time = 40
- acquisition time = 30
- number of repeated attacks = 20
- assignment protocol = 1

To customize the program, the user could change these default values in the program by altering the program statements in the AIRDEF.BAS program in lines 12000 to 12200. The best way to do so, is by listing these lines inside the basic environment by typing 'list 12000-12200' <ENTER> and, in the full screen editor, moving the cursor to the desired line, changing the value and pressing the <ENTER> key again. To run the simulation with the new default values, type 'run' <ENTER>.

2. <u>Default Parameter Protocol 2</u>

If this option is selected by pressing <2>, then the simulation starts with the same default simulation parameters as before when protocol 1 was

chosen, with the exception, that now assignment protocol 2 is used to make controlled target assignments.

3. Changing the Simulation Parameter

When <C> is pressed, AIRDEF will ask for the desired values of the parameters as shown in Figure 3. If the user tries to change the parameters to values outside of the given boundaries, the program will

```
              CHANGE SIMULATION PARAMETERS

Enter # of Attackers    (1...9): 5
Enter # of Defenders    (1...7): 5
Enter P[ hit a plane ] (0...1):.5
Enter Exposure time   (30..200):50
Enter Acquisitiontime (0..200):10
Enter # of repeated attacks    :10
Enter Protocol type   (1 or 2): 2
```

Figure 3. Change Simulation Parameter Screen

refuse to accept these values and ask for the same parameter value again. After entering the new values the <ENTER> key must be pressed every time. When all values are changed, the simulation starts a new run.

4. Setting the Random Number Generator Seed

A new seed for the random number generator can be entered, by pressing <N>. The program will prompt for a seed value as shown in Figure 4. Allowed seed values are only integers in the range from one to 2,147,483,646 (which is $2^{31} - 2$). Non-integers are truncated by the program.

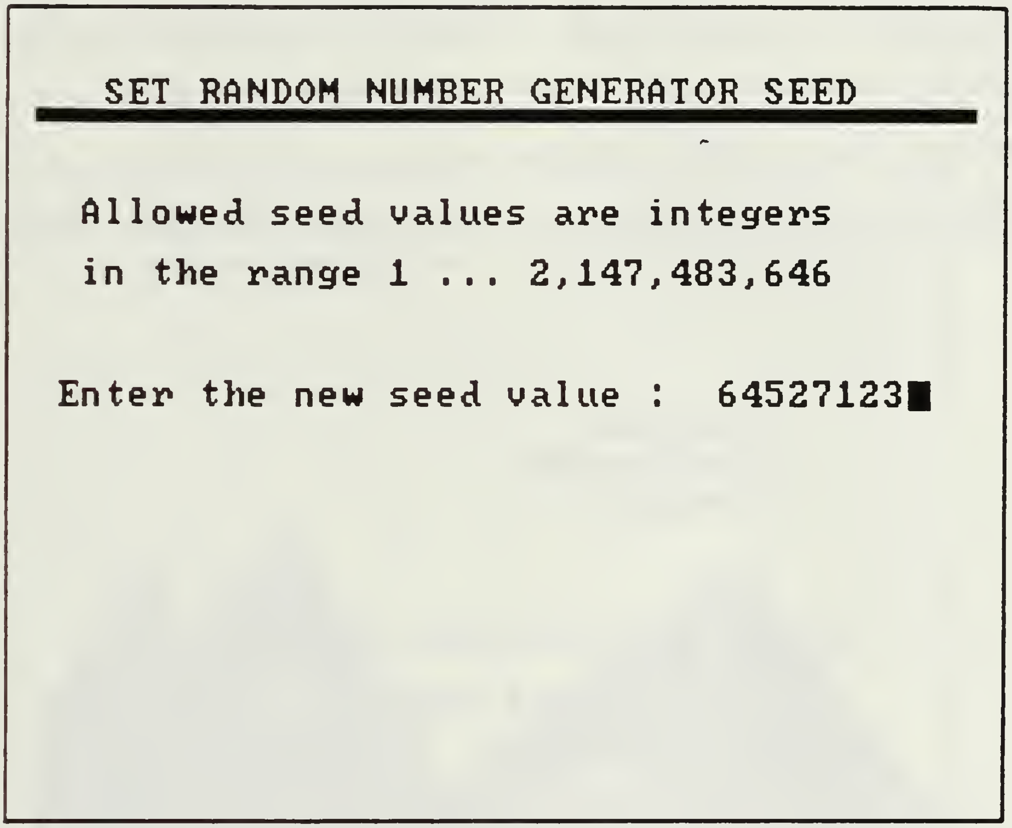

Figure 4. Set New Random Number Generator Seed

If no seed is selected, a default seed is used by the RNGEN.SRT and updated with each call for a random number. Every time the simulation is started from BASICA by typing 'run', the seed is set to its

initial value. If the simulation is started without leaving AIRDEF, the sequence of random numbers is not broken, even when a new run from the main menu is made.

5. Ending the program

To quit the program press <Q> at the main menu, at one of the statistics screens or at the iterations option screen. This will clear the screen and by showing the message in Figure 5, leave the PC in the BASICA command mode at 40 character width. If it is desired to go to 80 character width, type 'screen 2' <ENTER>, and to return to DOS, type 'system' <ENTER>.

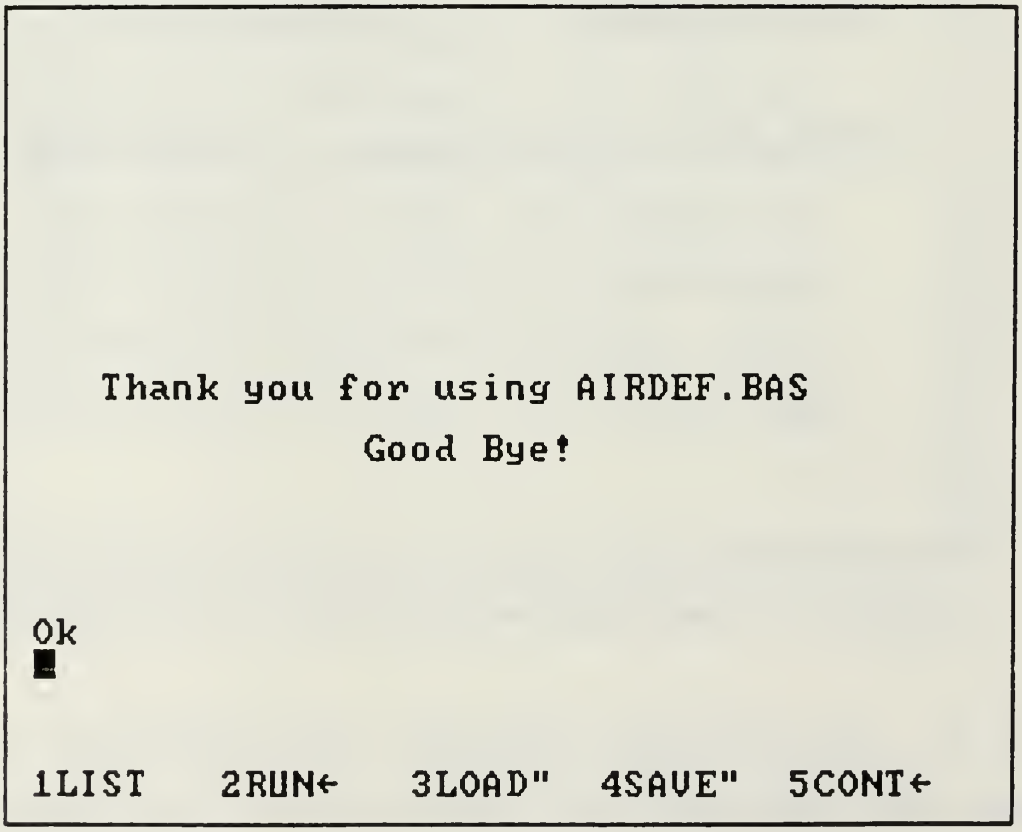

Figure 5. Ending Screen

F. THE GRAPHICS DISPLAYS

1. The Scenario Graphics Screen

AIDEF uses a graphics screen to show the animation of the simulation as seen in Figure 6 and Figure 7. Both assignment protocols, 1 and 2, work on the same type of graphics screen, only the action differs. To indicate which protocol is used, it is stated on the bottom line of the screen. The actual number of the attack flown is indicated in the lower right corner. The airplane formation flies from right to left and its speed is determined by the exposure time.

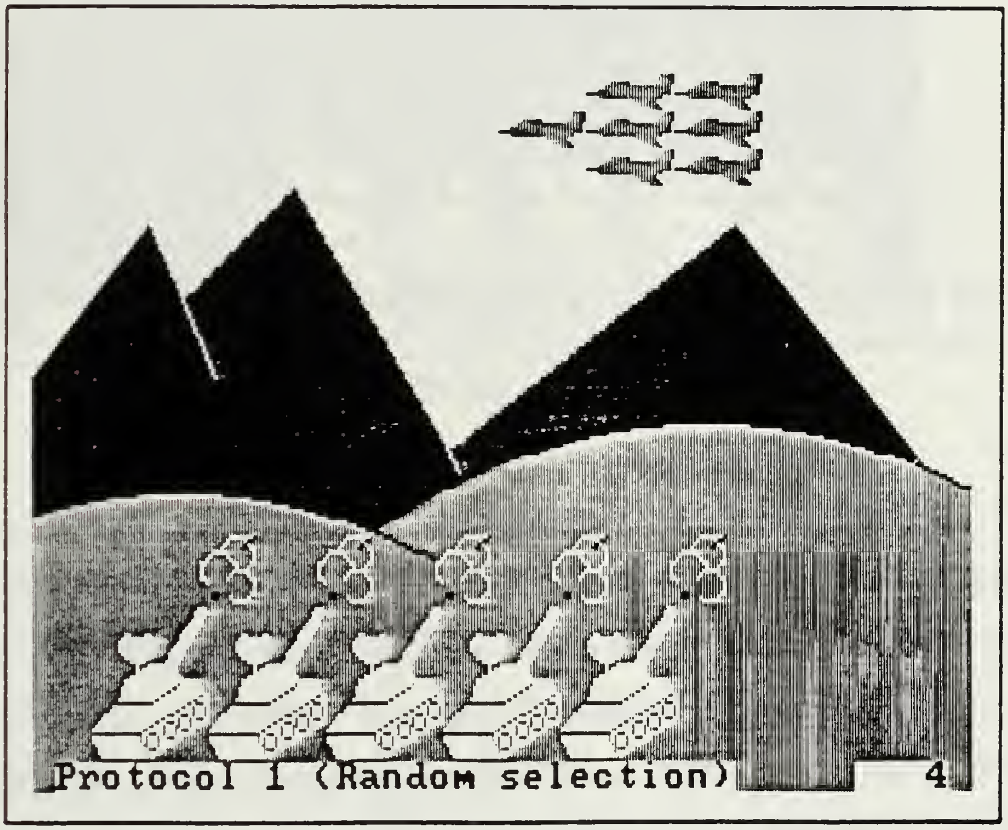

Figure 6. Graphics Screen Protocol 1

A shot is indicated by a small cloud above the ADA tanks. A hit is indicated by a fireball in place of the airplane. Even if an attacker is already killed, a multiple hit is also indicated as a fireball. (Multiple hits are only possible with protocol 1).

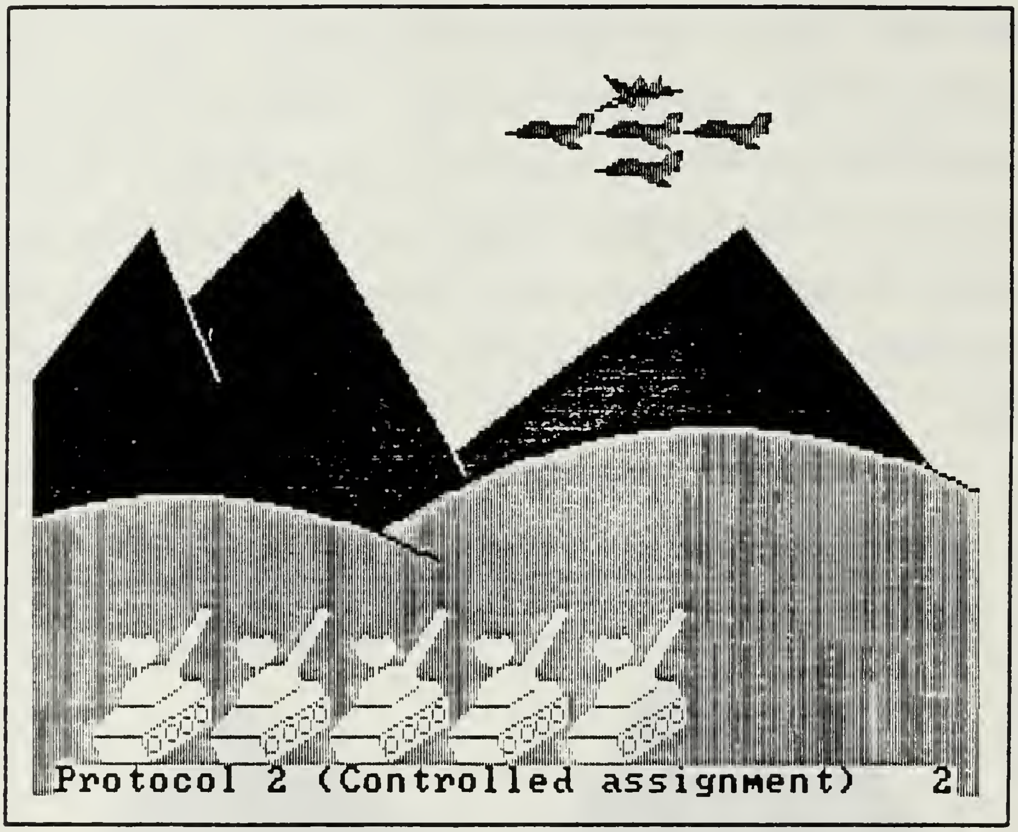

Figure 7. Graphics Screen Protocol 2

After the desired number of attacks are completed, or when any key was pressed, the simulation will stop at the end of the last attack and give a choice of two different types of statistical displays: numerical or graphical. Both present the same information, but in a different way.

2. <u>The Numerical Statistics Screen</u>

To receive the numerical statistics display, press <N> and the screen will clear and yield a numerical presentation of the simulation statistics as shown in Figure 8.

```
          PROTOCOL 1 with   20 ATTACKS
          ----------------------------
Attackers  : 7        Exposure time: 40
Defenders  : 5        Acquisit.time: 30
Hitprob.   :.20
--------------------------------------

Avg No of shots/attack          = 36.25

Avg No of kills/attack          =  6.80

Avg No of mult.hits/attack      =  0.50

Avg No of survivors/attack      =  0.20

Avg No of kills/shot            =  0.188

Avg time to last  survivor      = 37.25


<G>raphical statistic <C>ontinue <Q>uit
```

Figure 8. Numerical Statistics

The upper section shows the given simulation parameters, such as the type of protocol, the number of attacks actually flown, the number of attackers and defenders, the probability of hitting an aircraft, the exposure and the acquisition time. The rest of the screen shows all different kinds of averages computed

in the simulation. For a complete description of the implications and possible values of the displayed averages see chapter B. MODEL ESTIMATORS on pages 12 to 15, earlier in this paper.

A quick crosscheck of the values can be made, by ensuring, that the sum of the averages of kills per attack and survivors per attack add up to the number of attackers, or checking that the average of kills per attack divided by the average of shots per attack equals the average of kills per shot, or checking that the value of the average time to the last survivor is higher than the acqusition time, but lower or equal to the exposure time.

On the bottom line of the screen a small menu gives the user the options of going to the graphical representation of the statistics, continuing to the next menu or quitting the program.

Quitting the program by pressing the <Q> key will exit the program and lead the PC into the BASICA command mode as shown in Figure 5.

Continuing the program by pressing <C> will cause the iterations option screen, as described in the capital F. 5. on page 49 and shown in Figure 11, to be displayed.

Pressing <G> will change the screen to the graphical statistics screen (Figure 9 or Figure 10).

3. The Graphical Statistics Screen

When this screen is evoked the first time in the simulation, the graphical representation utilizes only the left half of the display as shown in Figure 9.

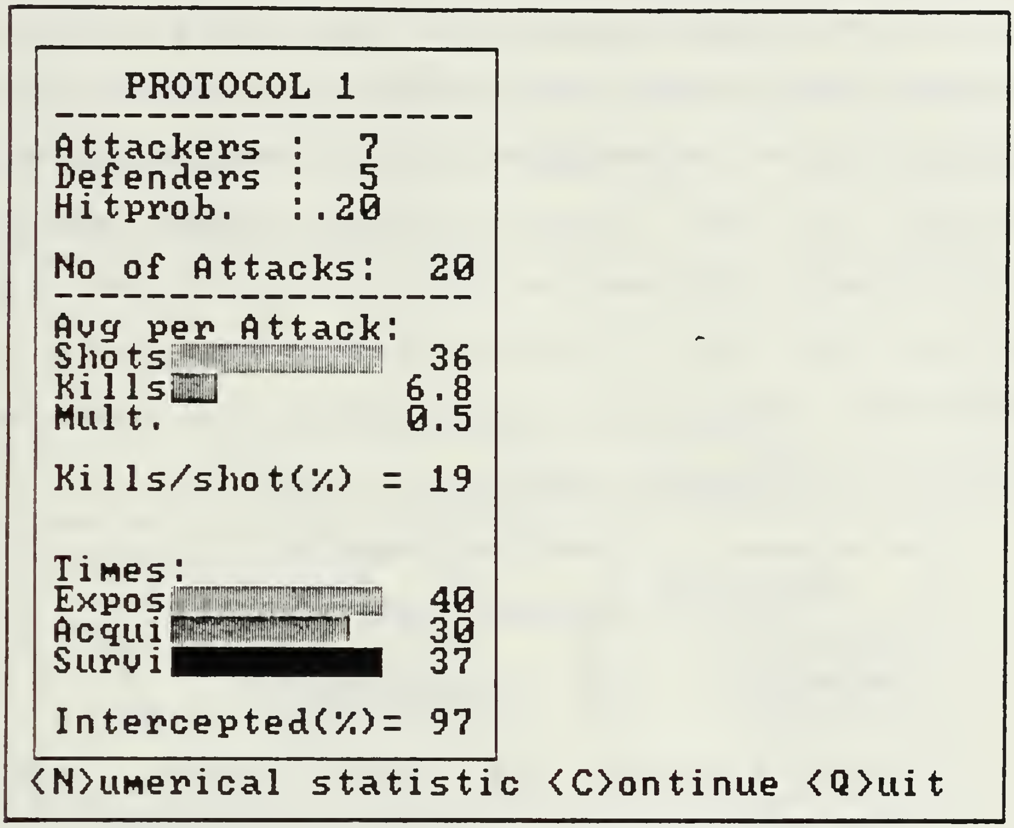

Figure 9. Graphical Statistics (half)

This is the graphical representation of the numerical statistical values. The upper section shows part of the desired model parameters as before, and the averages are visualized as horizontal bargraphs. The user is enabled to compare the values graphically. The percentage of kills per shot and successfully intercepted attacks are shown numerically.

The line at the bottom of the screen offers the user the option to switch back to the numerical statistics screen, to continue to the iterations option screen or to quit the program.

Entering the graphical statistics screen option a second time will result in the graphical utilization of the full screen (Figure 10). The bargraphs, formerly on the left half of the graphical screen, are shifted to the right side and the new statistics are displayed on the left side. Thus the user is enabled to compare the former statistical results with the most recently computed simulation results.

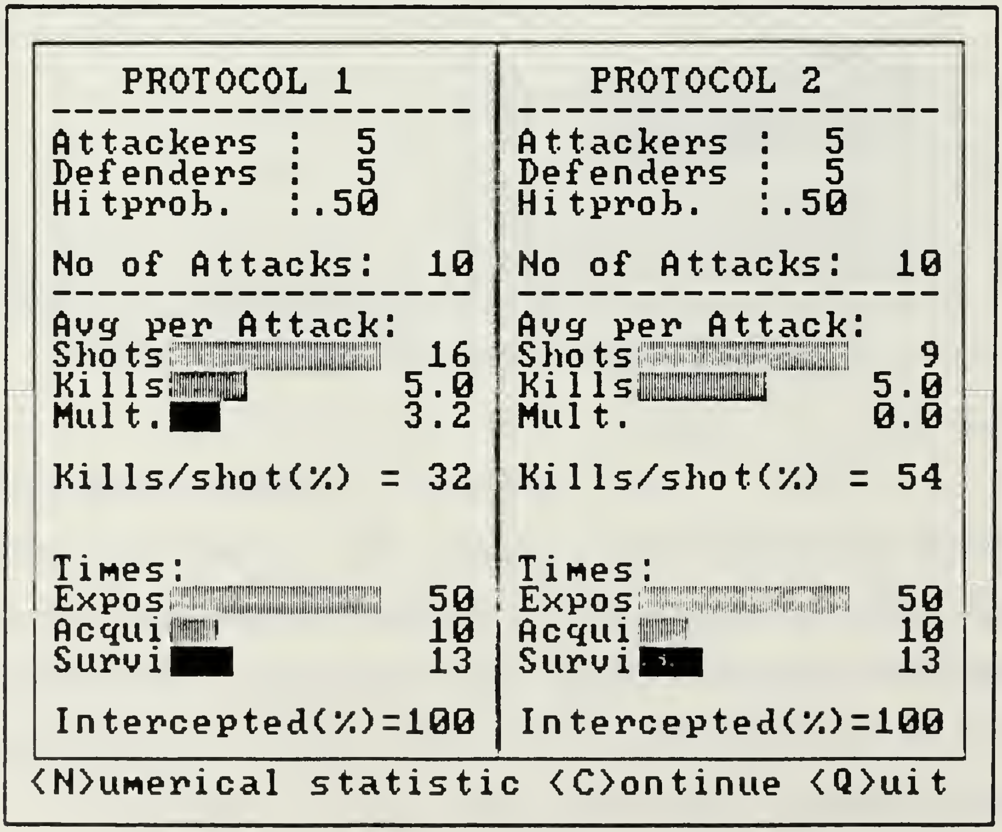

Figure 10. Graphical Statistics (full)

4. <u>The Iterations Option Screen</u>

Pressing the <C> key will display the iterations option screen in Figure 11. This allows the user to proceed in the program without going back to the main menu. Basically, it is intended to start new iterations in the simulation, which have the same parameters as before, but using another type of target assignment protocol.

```
New Iterations under:
<S>ame Protocol
<O>ther Protocol

<M>ain menu
<Q>uit

Enter your selection:

<N>umerical statistic <C>ontinue <Q>uit
```

Figure 11. Iterations Option Screen

Pressing <S> will start new attacks with the same type of protocol. Pressing <O> will start them with the alternative protocol. The random number

generator seed will not be reset. This is only the case when the program is restarted from the BASIC command level. Hereby the user is able to compare simulation runs with different seed values, without changing the seed from the main menu.

In addition to this, the user will still have the option of going back to the main menu or quitting the program.

G. PROGRAM MODIFICATIONS

To alter the program, the user should consult a BASIC programming manual of how to change and save the modified program. A backup copy of the original is essential at this point, otherwise a well running program could be lost.

Permanent changes could be made for the default values of the simulation parameters. The default values are set in the program section lines 12000 - 12200.

Eliminating the beep, everytime a hit occurs, could be done by deleting line 51260 and line 52180.

After studying the program structure, an experienced programmer could add additional types of assignment protocols to the model, since the program, structured with subroutines, is open for changes.

APPENDIX B

AIRDEF.BAS PROGRAM LISTING

```
1  '***************************************************
2 '*                                                   *
3 '* AIRDEF.BAS  V1.0,   Klaus Schuster,  Aug 1987     *
4 '*                                                   *
5 '**************************************************
6 '
10 CLEAR,,16384 : KEY OFF : CLS
97 '
98 '************************  Initialisation  ********
99 '
100 DIM PLANE%(124),TANK%(496),BULL%(10),BUM%(100)
105 DIM CLOUD%(100),BLANK%(100),FORMATION%(2000)
115 DIM STAT1%(8010),PT(8),PLNR(9)
220 GOSUB 5000    'Check for BASICA and color/graphics
230 GOSUB 6000                   'Print Title screen
250 GOSUB  10000          'Load random number generator
260 GOSUB 11000               'Print main program menu
799 '
800 '*****************************  new run  ********
801 '
810 REPEAT = REPEAT + 1
860 SCREEN 1
865 COLOR 1,0
867 CLS
897 '
898 '*********************  draw green hills  ********
899 '
900 CIRCLE(50,300),200,1,.93,2!,.9
905 CIRCLE(230,300),220,1,1!,2.12,.9
910 PAINT(30,180),1
912 '
913 '*****************  draw brown mountains  ********
914 '
915 CIRCLE(50,300),200,3,1.1,2!,.9
920 CIRCLE(230,300),220,3,1!,2.12,.9
930 LINE(0,90)-(40,50),3
935 LINE(40,50)-(63,90),3
940 LINE(52,70)-(90,40),3
945 LINE(90,40)-(146,115),3
950 LINE(142,110)-(240,50),3
955 LINE(240,50)-(304,114),3
960 PAINT(30,70),3
970 LINE(52,70)-(63,90),0
980 LINE(142,109)-(146,115),0
995 IF REPEAT > 1 GOTO 1800
```

```
998 '
999 '**********************  draw red plane  ********
1000 '
1001 LINE (5,8)-(14,8),2
1002 LINE (22,11)-(14,8),2
1004 LINE (22,11)-(25,11),2
1006 LINE (22,8)-(25,11),2
1008 LINE (22,8)-(28,7),2
1010 LINE (29,2)-(28,7),2
1016 LINE (29,2)-(25,2),2
1018 LINE (22,5)-(25,2),2
1022 LINE (22,5)-(8,4),2
1024 LINE (6,5)-(8,4),2
1026 LINE (6,5)-(2,7),2
1028 LINE (0,7)-(2,7),2
1030 LINE (5,8)-(2,7),2
1040 PAINT(16,7),2
1050 GET (0,2)-(29,11),PLANE%
1055 PUT (0,2),PLANE%
1097 '
1098 '**********************  draw blue tank  ********
1099 '
1100 LINE(24,190)-(40,190),0
1110 LINE(60,180)-(40,190),0
1120 LINE(60,180)-(60,174),0
1130 LINE(56,170)-(60,174),0
1140 LINE(56,170)-(50,170),0
1150 LINE(49,167)-(50,170),0
1160 LINE(46,166)-(40,166),0
1170 LINE(32,170)-(40,166),0
1180 LINE(32,170)-(30,176),0
1190 LINE(24,180)-(30,176),0
1200 LINE(24,180)-(20,184),0
1210 LINE(24,190)-(20,184),0
1220 LINE(36,168)-(36,166),0
1230 LINE(48,168)-(60,152),0
1245 LINE(45,166)-(58,150),0
1250 LINE(60,152)-(58,150),0
1255 CIRCLE(36,154),6,0,4.1,5.3,.8
1260 CIRCLE(36,161),6,0,2.36,.785,.7
1270 PAINT(40,175),0
1280 PAINT(36,162),0
1290 LINE(30,176)-(40,176),1
1300 LINE(50,171)-(40,176),1
1310 LINE(20,184)-(36,184),1
1320 LINE(40,190)-(36,184),1
1330 LINE(60,173)-(36,184),1
1340 CIRCLE(41,186),2,1
1350 CIRCLE(47,183),2,1
1360 CIRCLE(52,181),2,1
1370 CIRCLE(57,178),2,1
```

```
1390 GET (20,150)-(60,190),TANK%
1398 '
1400 '*************************  draw cloud   ********
1401 '
1405 GET (2,131)-(20,150),BLANK%
1410 CIRCLE (102,141),6,0,0,4
1420 CIRCLE (109,145),5,0,4,2.5
1430 CIRCLE (108,137),6,0,5.5,2.6
1440 GET (96,131)-(114,150),CLOUD%
1445 PUT (96,131),BLANK%,PSET
1499 '
1500 '*********************  draw red bullet  ********
1501 '
1505 LINE (98,20)-( 99,21),2,BF
1510 GET (98,20)-( 99,21),BULL%
1520 PUT (98,20),BULL%
1619 '
1620 '**************  draw red/brown fireball ********
1621 '
1630 PSET (3,0)
1640 DRAW "p2,3"
1650 DRAW "f5;m0,9;m10,7;f2;e2;f2;e3;m29,4"
1660 DRAW "M22,3;u3;m18,3;h3;g3;m3,0"
1670 PAINT (15,5),2
1680 PSET (10,4)
1690 DRAW "p3,3;m13,6;e1;r6;m15,3;g2;m10,4"
1710 PAINT (15,4),3
1730 GET (0,0)-(29,9),BUM%
1750 PUT (0,0),BUM%
1799 '
1800 '***************  put T tanks on screen  ********
1801 '
1805 FOR I=1 TO T
1810 X=20+40*(I-1)
1820 PUT (X,150),TANK%,AND
1830 NEXT I
1899 '
1900 '*************************  write footer  ********
1901 '
1910 LOCATE 25,2
1920 IF SIM = 1 THEN PRINT USING "Protocol #
                    (Random selection)";SIM;
1930 IF SIM = 2 THEN PRINT USING "Protocol #
                    (Controlled assignment)";SIM;
1999 '
2000 '***********************  animate plane  ********
2001 '
2002 LOCATE 25,36: PRINT USING "####";RUNCOUNT+1;
2005 X = 200:Y = 10          'init position of formation
2010 FOR I=1 TO PL      'get positions of single planes
```

```
2020 ON I GOSUB 5100,25200,25300,25400,25500,
                25600,25700,25800,25900
2030 PUT (XP,YP),PLANE%             'put plane on screen
2040 NEXT I
2050 GET (200,10)-(319,39),FORMATION% 'store formation
2060 PUT(200,10),FORMATION%          'remove formation

2062 FOR I = 1 TO PL:PLNR(I) = 500: NEXT    'init plane
                                        not shot = 500
2065 FOR  I = 1 TO T:PT(I) = I: NEXT 'init assign plane
                                               to tank
2066 RUNHIT = 0: ROUND = 0: AS=T: L=T             'init
2067 IF SIM=2 AND PL<T THEN : L=PL
2068 RUNCOUNT = RUNCOUNT + 1                'count runs
2070 FOR X=200 TO 1 STEP -PLSPEED      'move formation
2080 PUT (X,10),FORMATION%,PSET
2085 IF X>200-ALLOC THEN :FOR DELAY=1 TO 100 :NEXT:
                      GOTO 2310 'wait with shoting
2088 ROUND=ROUND+1  :ROUNDNR=ROUNDNR+1   'count rounds
2089 IF SIM = 2 THEN : GOSUB 52000
2090 IF SIM = 1 THEN : GOSUB 51000
2305 IF RUNHIT >= PL GOTO 2500 'if all planes hit,
                                             end round
2310 NEXT X
2320 FOR I=1 TO PL
2330 IF PLNR(I)=500 THEN:SURV=SURV+1   'count survivor
2340 NEXT I
2350 PUT (X+PLSPEED,10),FORMATION% 'remove formation on
                                         end  of screen
2500 IF INKEY$ <> "" GOTO 2510 'if any key hit, end
                                                 round
2505 IF  RUNCOUNT < RUNNR GOTO 2000 'if not all attacks
                                        done, new run
2506 FOR DELAY = 1 TO 1000: NEXT        'wait a second
2507 '
2508 '********************  end of animation  ********
2509 '
2510 SOUND 90,2:SOUND 200,2:SOUND 90,2:SOUND 200,2
2515 LOCATE 1,1
2520 PRINT "<N>umerical or <G>raphical statistics"
2525 GOSUB 20000
2530 IF RESPONSE$="n" OR RESPONSE$="N" THEN : GOSUB
                                              30000
2535 IF  RESPONSE$="g" OR RESPONSE$="G" THEN : GOSUB
                                              35000
2547 CLS
2548 LINE (78,45)-(258,147),2,B
2550 LOCATE 8,12
2552 PRINT "New Iterations under:"
2554 LOCATE 10,12
2555 PRINT "<S>ame Protocol"
```

```
2557 LOCATE 12,12
2558 PRINT "<O>ther Protocol"
2559 LOCATE 15,12
2560 PRINT "<M>ain menu"
2561 LOCATE 17,12 : PRINT "<Q>uit"
2562 LOCATE 21,12 : PRINT "Enter your selection:"
2563 GOSUB 20000
2564 IF RESPONSE$="s" OR RESPONSE$="S" GOTO 2570
2565 IF RESPONSE$="o" OR RESPONSE$="O" THEN:IF  SIM=1
                    THEN   SIM=2:ELSE  SIM=1:GOTO  2570
2566 IF RESPONSE$="q" OR RESPONSE$="Q" GOTO 40000
2570 SURV=0 : HIT=0 : ROUNDNR=0 : SHOT=0 : DBL=0 :
                                     RUNCOUNT=0    'reset
2579 IF RESPONSE$="m" OR RESPONSE$="M" GOTO 11000
2580 GOTO 800                               'start new runs
4999 '
5000 '******  BASICA and Colorgraphics check  ********
5001 '
5010 DEF SEG = 0
5020 IF  (PEEK(&H410) AND &H30) <> &H30 THEN DEF SEG :
                                                GOTO 5070
5030 LOCATE 3,1:PRINT "Sorry ... "
5040 PRINT "you do not have  the color/graphics monitor
                                                 adapter!"
5050 PRINT  "This simulation uses graphics and requires
                                             that adapter."
5060 DEF SEG:END
5070 ON ERROR GOTO 5080 : PLAY "p16" : GOTO 5130
5080 WIDTH 80 : CLS : LOCATE 3,1
5090 PRINT "This simulation uses advanced BASIC."
5100 PRINT "Return to DOS by typing 'SYSTEM',"
5110 PRINT "load BASICA and reload this program"
5120 END
5130 ON ERROR GOTO 0
5140 RETURN
5999 '
6000 '************************  Title Screen  ********
6001 '
6010 SCREEN 0,1  : COLOR 14,1,1 : WIDTH 40 : CLS
6020 LOCATE 2,12 : PRINT "GRAPHIC SIMULATION"
6030 LOCATE 4,19 : PRINT "of an"
6040 LOCATE 6,11 : PRINT "AIR DEFENSE SCENARIO"
6050 LOCATE 9,13 : PRINT "by Klaus Schuster"
6060 LOCATE 14,5 : PRINT "Submitted in partial
                                      fulfillment"
6070 LOCATE  15,2 : PRINT "of the requirements for the
                                            degree of"
6080 LOCATE 17,1 : PRINT "Master of Science in
                              Operations Research"
6090 LOCATE  19,4 : PRINT "from the Naval Postgraduate
                                               School"
```

```
6100 LOCATE 20,11: PRINT "Monterey, California"
6110 LOCATE 22,6 : PRINT "Advisors J.D. Esary, P.
                                            Purdue"
6120 LOCATE  25,2 : PRINT "Press any key to
                                 continue...";
6130 GOSUB 20000
6140 RETURN
6999 '
7000 '*************************  Instructions  ********
7001 '
7010 SCREEN 0,1 : COLOR 14,1,1 : WIDTH 40 : CLS
7020 LOCATE 3,6  : PRINT "INSTRUCTIONS FOR THE MENU"
7025 PRINT BORDER$
7030 LOCATE 5,2
7031 PRINT "  This program is an animated graphic"
7035 LOCATE 6,2
7036 PRINT "simulation of an air defense scenario"
7040 LOCATE 7,2
7041 PRINT "in which some attacking aircraft are"
7045 LOCATE 8,2
7046 PRINT "intercepted by several defenders."
7050 LOCATE 10,2
7051 PRINT "  Two target assignment protocols are "
7055 LOCATE 11,2
7056 PRINT "possible:"
7060 LOCATE 13,2
7061 PRINT "  (1) random selection by the defender"
7065 LOCATE 14,2
7066 PRINT "  (2) controlled target assignment"
7070 LOCATE 18,2
7071 PRINT "  The program menus will guide you."
7080 LOCATE 25,1 :
     PRINT "Press any key to continue ...";
7085 GOSUB 20000
7090 CLS
7100 LOCATE 1,2
7101 PRINT "Default parameters for both protocols:"
7105 LOCATE 3,2
7106 PRINT " Number of attackers       :   7"
7110 LOCATE 4,2
7111 PRINT " Number of defenders       :   5"
7115 LOCATE 5,2
7116 PRINT " Probability of hit        :  .2"
7120 LOCATE 6,2
7121 PRINT " Exposure time             :  40"
7125 LOCATE 7,2
7126 PRINT " Acquisition time          :  30"
7130 LOCATE 8,2
7131 PRINT " Number of repeated attacks:  20"
7145 LOCATE 11,2
7146 PRINT "Options of the main menu:"
```

```
7147 LOCATE 13,2
7148 PRINT "Pressing <I> gives the above"
7149 LOCATE 14,2
7150 PRINT "                instruction screens"
7151 LOCATE 15,2
7152 PRINT "Pressing <1> starts simulation with"
7155 LOCATE 16,2
7156 PRINT "                defaults in protocol 1"
7160 LOCATE 17,2
7161 PRINT "Pressing <2> starts simulation with"
7165 LOCATE 18,2
7166 PRINT "                defaults in protocol 2"
7170 LOCATE 19,2
7171 PRINT "Pressing <C> allows alteration of"
7175 LOCATE 20,2
7176 PRINT "                simulation parameters"
7180 LOCATE 21,2
7181 PRINT "Pressing <N> allows alteration of"
7185 LOCATE 22,2
7186 PRINT "                random number seed"
7197 LOCATE 23,2
7198 PRINT "Pressing <Q> quits the program"
7200 LOCATE 25,1 :
     PRINT "Press any key to continue ...";
7201 GOSUB 20000
7202 CLS
7205 LOCATE 1,2
7206 PRINT "The Statistics Screens:"
7210 LOCATE 3,2
7211 PRINT "To show the statistical results,"
7215 LOCATE 4,2
7216 PRINT "the program has two options:"
7220 LOCATE 6,2
7221 PRINT "  <N> Numerical statistics screen"
7225 LOCATE 7,2
7226 PRINT "  <G> Graphical statistics screen"
7230 LOCATE 9,2
7231 PRINT "Both present the same information."
7235 LOCATE 11,2
7236 PRINT "Upon entering the graphical screen a"
7240 LOCATE 12,2
7241 PRINT "second time, the previous screen will"
7245 LOCATE 13,2
7246 PRINT "be shown on the right side."
7250 LOCATE 15,2
7251 PRINT "Continuing from the statistics screens"
7255 LOCATE 16,2
7256 PRINT "leds to the iterations option screen"
7260 LOCATE 18,2
```

```
7261 PRINT "Pressing <S> starts simulation under "
7265 LOCATE 19,2
7266 PRINT "                 same default parameters"
7270 LOCATE 20,2
7271 PRINT "                 as before"
7275 LOCATE 21,2
7276 PRINT "Pressing <O> starts simulation under "
7280 LOCATE 22,2
7281 PRINT "                 other protocol than before"
7285 LOCATE 23,2
7286 PRINT "Pressing <M> goes back to main menu "
7290 LOCATE 24,2
7291 PRINT "Pressing <Q> quits the program";
7300 LOCATE 25,1 :
     PRINT "Press any key to go back to main menu";
7301 GOSUB 20000
7302 CLS
7500 GOTO 11010   'return to main menu
9999 '
10000 '*******  load random number generator  ********
10001 '
10010 RNGEN=0:U!=0
10015 ON ERROR GOTO 10050 'file not found
10020 DEF SEG = &H1A00
10030 BLOAD DRIVE$+"rngen.srt",0
10032 A# = 3245678!                   'default seed
10034 GOTO 10570
10040 RETURN
10050 IF ERR <>53 THEN ON ERROR GOTO 0
10060 CLS:LOCATE 3,1
10070 INPUT "On which drive could RNGEN.SRT be
                                    found?";A$
10080 DRIVE$ =A$+":"
10090 RESUME 10020
10499 '
10500 '**********  Change random number seed  ********
10501 '
10510 COLOR 14,0,0 : CLS
10520 LOCATE  3,4 : PRINT "SET RANDOM NUMBER GENERATOR
                                  SEED":PRINT BORDER$
10530 LOCATE 7,3 : PRINT "Allowed seed values are
                                        integers"
10540 LOCATE  9,3 : PRINT "in the range 1 ...
                              2,147,483,646"
10550 LOCATE 13,2 : INPUT "Enter the new seed value  :
                                             ",A#
10560 IF  A#<1 OR A#>2147483646# THEN LOCATE 13,2:PRINT
                          STRING$(39,32):GOTO 10550
10570 DEF SEG = &H1A00
10580 A1! = INT(A#/16777210#):
      A2! = INT((A#-A1!*16777210#)/65536!)
```

```
10600 'poke the seed's upper 2 bytes into  RNGEN's seed
                                                  storage
10610 POKE &H164,A1! : POKE &H163,A2!
10620 A# = A#-A1!*16777210#-A2!*65536!
10630 A1! = INT(A#/256) :
      A2! = A#-A1!*256
10640 'poke  the seed's lower 2 bytes into RNGEN's seed
                                                  storage
10650 POKE &H162,A1! : POKE &H161,A2!
10660 GOTO 11010   'return to main menu
10999 '
11000 '*******************  Main Program Menu  ********
11001 '
11005 INIT = 0
11010 BORDER$ = STRING$(40,223)
11020 SCREEN 0,1 : COLOR 14,3,0 : WIDTH 40 : CLS
11030 LOCATE 3,14 : PRINT "PROGRAM MENU" :
                    PRINT BORDER$
11040 LOCATE 5,3  : PRINT "<I>nstructions"
11050 LOCATE 7,3  : PRINT "<1>Default Parameter
                                             Protocol 1"
11060 LOCATE  9,3  : PRINT "<2>Default Parameter
                                             Protocol 2"
11070 LOCATE 11,3 : PRINT "<C>hange Simulation
                                              Parameter"
11075 LOCATE  13,3 : PRINT "<N>ew random number
                                         generator seed"
11080 LOCATE 15,3 : PRINT "<Q>uit Program"
11090 LOCATE 18,3 : PRINT "Enter your selection:";
11100 GOSUB 20000               'get keyboard response
11110 A% = INSTR("IN12CQin12cq",RESPONSE$)
11120 IF A% = 0 THEN 11010 : ELSE ON A% GOTO 7000,
                   10500,12000,12000,13000,40000,
              7000,10500,12000,12000,13000,40000
11999 '
12000 '*******  Default simulation parameter  ********
12001 '
12010 T = 5
12020 PL = 7
12030 HITPROB = .2
12050 EXPOS =  40    '30...200
12060 RUNNR = 20
12070 AIMTIME = 30
12080 SIM = 1
12085 IF A% = 4 OR A% = 10 THEN SIM = 2
12090 ALLOC = (200*AIMTIME)/EXPOS
12100 PLSPEED = 200/EXPOS
12200 GOTO 800
12999 '
```

```
13000 '*******  Change simulation parameters  ********
13001 '
13010 SCREEN 0,1 : COLOR 15,5,0 : CLS
13050 LOCATE 3,6 : PRINT "CHANGE SIMULATION PARAMETERS"
13060 PRINT BORDER$
13065 MSG1$ = "Can not be accepted! Please try again."
13070 LOCATE  6,1 : INPUT "Enter # of Attackers
                                              (1...9): ",PL
13080 IF PL<1 OR PL>9 THEN LOCATE 23,1: PRINT MSG1$:
                      LOCATE 6,1:PRINT STRING$(40,32):
                                                GOTO 13070
13090 LOCATE 23,1 : PRINT STRING$(40,32);
13100 LOCATE 8,1 :
            INPUT "Enter # of Defenders   (1...7): ",T
13110 IF T<1 OR T>7 THEN LOCATE 23,1:PRINT MSG1$:
                      LOCATE 8,1:PRINT STRING$(40,32):
                                                GOTO 13070
13120 LOCATE 23,1 : PRINT STRING$(40,32);
13130 LOCATE 10,1 : INPUT "Enter P[ hit a plane ]
                                          (0...1):",HITPROB
13140 IF HITPROB<0 OR HITPROB>1 THEN LOCATE 23,1:
         PRINT MSG1$:LOCATE 10,1:PRINT STRING$(40,32):
                                                GOTO 13130
13150 LOCATE 23,1 : PRINT STRING$(40,32);
13160 LOCATE  12,1: INPUT "Enter Exposure time
                                          (30..200):",EXPOS
13170 IF EXPOS<30 OR EXPOS>200 THEN LOCATE 23,1:
         PRINT MSG1$:LOCATE 12,1:PRINT STRING$(40,32):
                                                GOTO 13160
13180 LOCATE 23,1 : PRINT STRING$(40,32);
13190 LOCATE 14,1: INPUT "Enter Acquisitiontime
                                          (0..200):",AIMTIME
13200 IF AIMTIME<0 OR AIMTIME>200 THEN LOCATE 23,1:
         PRINT MSG1$:LOCATE 14,1:PRINT STRING$(40,32):
                                                GOTO 13190
13210 LOCATE 23,1 : PRINT STRING$(40,32);
13220 LOCATE 16,1: INPUT "Enter # of repeated attacks :
                                                  ",RUNNR
13230 LOCATE  18,1: INPUT "Enter Protocol type   (1 or
                                                2): ",SIM
13240 IF SIM<1 OR SIM>2 THEN LOCATE 23,1:PRINT MSG1$:
                     LOCATE 18,1:PRINT STRING$(40,32):
                                                GOTO 13230
13250 LOCATE 23,1 : PRINT STRING$(40,32);
13260 ALLOC = (200*AIMTIME)/EXPOS
13270 PLSPEED = 200/EXPOS
13300 GOTO 800
19999 '
```

```
20000 '**************  Get keyboard response  ********
20001 '
20010 DEF SEG : POKE &H6A,0
20020 RESPONSE$ = INKEY$ : IF RESPONSE$ = "" THEN 20020
20030 RETURN
24999 '
25000 '**********  get coordinates for plane  ********
25001 '
25100 XP = X
25110 YP = Y + 10
25120 RETURN
25200 XP = X + 30
25210 YP = Y
25220 RETURN
25300 XP = X + 30
25310 YP = Y + 20
25320 RETURN
25400 XP = X + 60
25410 YP = Y + 10
25420 RETURN
25500 XP = X + 30
25510 YP = Y + 10
25520 RETURN
25600 XP = X + 60
25610 YP = Y
25620 RETURN
25700 XP = X + 60
25710 YP = Y + 20
25720 RETURN
25800 XP = X
25810 YP = Y
25820 RETURN
25900 XP = X
25910 YP = Y + 20
25920 RETURN
25999 '
30000 '***************  numerical statistics  ********
30001 '
30005 CLS
30010 VIEW SCREEN (1,8)-(318,168),0,2
30020 LOCATE 3,7:PRINT USING
         "PROTOCOL # with #### ATTACKS";SIM;RUNCOUNT
30022 LOCATE 4,7:PRINT
         "------------------------------"
30025 LOCATE 5,2
30030 PRINT USING
         "Attackers : #      Exposure time:###";PL;
                                                 EXPOS
30032 LOCATE 6,2
```

```
30035 PRINT USING
            "Defenders : #      Acquisit.time:###";T;
                                                 AIMTIME
30039 LOCATE 7,2
30040 PRINT USING "Hitprob.  :.##";HITPROB
30044 LOCATE 8,2
30045 PRINT
             "-------------------------------------"
30049 LOCATE 10,2
30050 PRINT USING
            "Avg No of shots/attack      =###.##";
                                            SHOT/RUNCOUNT
30059 LOCATE 12,2
30060 PRINT USING
            "Avg No of kills/attack      = ##.##";
                                             HIT/RUNCOUNT
30069 LOCATE 14,2

30070 PRINT USING
            "Avg No of mult.hits/attack  = ##.##";
                                             DBL/RUNCOUNT
30079 LOCATE 16,2
30080 PRINT USING
            "Avg No of survivors/attack  = ##.##";
                                            SURV/RUNCOUNT
30089 LOCATE 18,2
30090 PRINT USING
            "Avg No of kills/shot        =  #.###";
                                                 HIT/SHOT
30099 LOCATE 20,2
30100 PRINT USING
            "Avg time to last  survivor  =###.##";
                         (ROUNDNR/RUNCOUNT) + AIMTIME
30150 VIEW
30180 LOCATE 25,1 :
      PRINT "<G>raphical statistic <C>ontinue <Q>uit";
30190 GOSUB 20000
30200 IF RESPONSE$="g" OR RESPONSE$="G" THEN CLS :
                                              GOSUB 35000
30210 IF RESPONSE$="q" OR RESPONSE$="Q" THEN :
                                              GOSUB 40000
30240 RETURN
34999 '
35000 '***************  graphical statistics  ********
35001 '
35010 SCREEN 1 : COLOR 1,0 : CLS
35020 IF GRPH = 1 THEN PUT (159,1),STAT1%
35025 LINE (1,1)-(158,189),2,B
35030 LOCATE 2,5 : PRINT USING"PROTOCOL #";SIM
35035 LOCATE 3,2 : PRINT      "------------------";
35040 LOCATE 4,2 : PRINT USING "Attackers :  #";PL
```

```
35050 LOCATE 5,2 : PRINT USING "Defenders :  #";T
35060 LOCATE 6,2 : PRINT USING "Hitprob.   :.##";HITPROB
35065 LOCATE 8,2 : PRINT USING "No of Attacks:####";
                                                   RUNCOUNT
35070 LOCATE 9,2 : PRINT       "------------------";
35075 LOCATE 10,2 : PRINT "Avg per Attack:";
35080 LOCATE 11,2 : PRINT "Shots"
35090 LOCATE 12,2 : PRINT "Kills"
35100 LOCATE 13,2 : PRINT "Mult."
35110 LINE (48,80)-(128,87),1,BF
35120 LINE (48,88)-(((80/SHOT)*HIT)+48,95),2,BF
35130 LINE (48,96)-(((80/SHOT)*DBL)+48,103),3,BF
35140 LOCATE 11,16 : PRINT USING "####";SHOT/RUNCOUNT
35150 LOCATE 12,16 : PRINT USING "##.#";HIT/RUNCOUNT
35160 LOCATE 13,16 : PRINT USING "##.#";DBL/RUNCOUNT
35170 LOCATE 15,2 : PRINT USING "Kills/shot(%) =###";
                                              100*HIT/SHOT
35175 LOCATE 18,2 : PRINT       "Times:"
35180 LOCATE 19,2 : PRINT "Expos"
35190 LOCATE 20,2 : PRINT "Acqui"
35200 LOCATE 21,2 : PRINT "Survi"
35210 LINE (48,144)-(128,151),1,BF
35220 LINE (48,152)-(((80/EXPOS)*AIMTIME)+48,159),2,BF
35230 LINE (48,160)-(((80/EXPOS)*((ROUNDNR/RUNCOUNT)+
                                 AIMTIME))+48,167),3,BF
35240 LOCATE 19,16 : PRINT USING "####";EXPOS
35250 LOCATE 20,16 : PRINT USING "####";AIMTIME
35260 LOCATE 21,16 : PRINT USING "####";
                             (ROUNDNR/RUNCOUNT)+AIMTIME
35270 LOCATE 23,2  : PRINT USING "Intercepted(%)=###";
                           100*(1-(SURV/(RUNCOUNT*PL)))
35275 GET (1,1)-(158,199),STAT1%
35277 GRPH = 1
35280 LOCATE 25,1 : PRINT "<N>umerical statistic
                           <C>ontinue <Q>uit";
35290 GOSUB 20000
35300 IF  RESPONSE$="n" OR RESPONSE$="N" THEN CLS :
                                              GOSUB 30000
35310 IF RESPONSE$="q" OR RESPONSE$="Q" THEN :
                                              GOSUB 40000
36000 RETURN
39997 '
39998 '***********************  Exit program  ********
39999 '
40000 CLS
40005 LOCATE12,4:PRINT"Thank you for using AIRDEF.BAS "
40010 LOCATE 14,15:PRINT"Good Bye!"
40011 FOR DELAY = 1 TO 1000: NEXT
40015 KEY ON
40025 LOCATE 20,1
40030 END
```

```
50999 '
51000 '********************** protocol sim 1  ********
51001 '
51020 FOR I=1 TO L       'put clouds for shooting tanks
51030 XC = 18+40*I
51040 XB = 22+40*I
51050 PUT (XC,131),CLOUD%,PSET
51060 PUT (XB,147),BULL%,PSET
51070 NEXT I
51080 FOR I=1 TO L                         'remove clouds
51090 XC = 18+40*I
51100 XB = 22+40*I
51110 PUT (XC,131),BLANK%,PSET
51120 NEXT I
51130 FOR I=1 TO T                        'for every shot
51140 SHOT = SHOT + 1                        'count shots
51150 DEF SEG = &H1A00
51160 CALL RNGEN(U!)                           'get rnd #
51170 IF U! > HITPROB GOTO 51320  'if no hit next tank
51180 CALL RNGEN(U!)
51190 TP = CINT(U!*PL + .5)
51200 IF PLNR(TP) < ROUND GOTO 51180
51210 IF PLNR(TP) = ROUND THEN:DBL=DBL+1:GOTO 51250 'if
                                 shot this round,dbl hit
51220 HIT = HIT + 1                     'count total hits
51230 RUNHIT = RUNHIT + 1        'count hits this round
51240 PLNR(TP) = ROUND           'set pl shot in round#
51250 ON TP GOSUB 25100,25200,25300,25400,25500,
            25600,25700,25800,25900
51260 SOUND 90,2                           'sound and bum%
51270 PUT (XP,YP),BUM%,PSET
51280 FOR DELAY = 1 TO 500 : NEXT
51290 PUT (XP,YP),BUM%
51300 GET  (X,10)-(X+119,39),FORMATION%       'get new
                              formation w/o shot planes
51320 NEXT I
51400 RETURN
51999 '
52000 '********************** protocol sim 2  ********
52001 '
52010 FOR I=1 TO L                    'put clouds to tank
52020 IF PT(I) = 0 GOTO 52100   'if  no assignment next
                                                  tank
52030 XC = 18+40*I
52040 XB = 22+40*I
52050 PUT(XC,131),CLOUD%,PSET
52060 PUT(XB,147),BULL%,PSET
52070 FOR DELAY = 1 TO 100 : NEXT
52080 PUT(XC,131),BLANK%,PSET                'remove cloud
52090 SHOT = SHOT + 1                         'count shots
52100 NEXT I
```

```
52110 FOR I = 1 TO L
52115 IF PT(I) = 0 GOTO 52260      'if no assignment next
                                                     tank
52120 DEF SEG = &H1A00
52130 CALL RNGEN(U!)                               'get rnd #
52140 IF U! > HITPROB GOTO 52260   'if no hit next tank
52150 HIT = HIT + 1                       'count total hits
52160 RUNHIT = RUNHIT + 1                       'count hits
52170 ON PT(I) GOSUB 25100,25200,25300,25400,25500,
               25600,25700,25800,25900
52180 SOUND 90,2                           'sound and bum%
52190 PUT (XP,YP),BUM%,PSET
52200 FOR DELAY = 1 TO 500 : NEXT
52210 PUT(XP,YP),BUM%                'remove bum and plane
52220 GET(X,10)-(X+119,39),FORMATION%          'get new
                              formation w/o shot planes
52225 PLNR(PT(I)) = ROUND        'set pl shot in round#
52230 PT(I) = 0        'set tank free for new assignment
52240 IF AS < PL THEN AS = AS + 1 : PT(I) = AS 'get new
                                   assignment for tank
52260 NEXT I
52270 RETURN
```

LIST OF REFERENCES

1. Davison, R.J., Graphic Simulation of the Poisson process, Masters Thesis, Naval Postgraduate School, Monterey, California, October 1982.

2. Nelsen, R.E., Graphic Simulation of a Machine-Repairman Model, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1984.

3. Greene, G.F., Graphic Simulation of a Jackson Network, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1986.

BIBLIOGRAPHY

Gaver, D.P., Analysis of Combat via Poisson Process Models, OA 4301 Stochastic Models II class handout, Naval Postgraduate School, Monterey, 12 January 1987.

Hearn, D. and Baker, M.P., Computer Graphics for the IBM Personal Computer, Prentice-Hall, Inc., 1983.

Hearn, D. and Baker, M.P., Computer Graphics, Prentice-Hall, Inc., 1986.

Microsoft, Corporation, GW-BASIC Interpreter, Users's Guide and User's Reference, 1986.

Morgan, B.J.T., Elements of Simulation, Chapmann and Hall Ltd., 1984.

Traister, R.J., Music & Speech Programs for the IBM PC, TAB BOOKs Inc., 1983.

Wolverton, V., Supercharging MS-DOS, Microsoft Press, 1986.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center — 2
   Cameron Station
   Alexandria, Virginia 22304-6145

2. Library, Code 0142 — 2
   Naval Postgraduate School
   Monterey, California 93943-5002

3. Professor P. Purdue, Code 55Pd — 2
   Department of Operations Research
   Naval Postgraduate School
   Monterey, California 93943-5000

4. Professor J.D. Esary, Code 55Ey — 2
   Department of Operations Research
   Naval Postgraduate School
   Monterey, California 93943-5000

5. Hauptmann Klaus Schuster — 2
   Materialamt der Luftwaffe
   Dezernat III B 6
   Postfach 902500/503
   D5000 Koeln 90
   West Germany